UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/772,518	02/05/2004	Robert E. Dye	5150-38605	5410

7590 01/20/2010

Jeffrey C. Hood
Meyertons, Hood, Kivlin, Kowert & Goetzel PC
P.O. Box 398
Austin, TX 78767

EXAMINER	
CHEN, QING	
ART UNIT	PAPER NUMBER
2191	
MAIL DATE	DELIVERY MODE
01/20/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/772,518
Filing Date: February 05, 2004
Appellant(s): DYE ET AL.

Jeffrey C. Hood (Reg. No. 35,198)
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal briefs filed on October 6, 2009 and November 20, 2009 appealing from the Office action mailed on June 4, 2009.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The Examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct. However, it is noted that Claims 59-104 stand rejected under 35 U.S.C. § 103(a) and not 35 U.S.C. § 102(b) as stated in the brief.

(4) Status of Amendments After Final

The Appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The Appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

4,901,221	KODOSKY et al.	2-1990
5,801,689	HUNTSMAN	9-1998

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

1. **Claims 59-104** are rejected under 35 U.S.C. 103(a) as being unpatentable over **US 5,801,689 (hereinafter "Huntsman")** in view of **US 4,901,221 (hereinafter "Kodosky")**.

As per **Claim 59**, Huntsman discloses:

- establish a network connection with a client computer system over a network *(see Column 9: 12-14, "The second computer 25 is connected to the first computer over the internet 31 ...")*;

- receive user input from the client computer system indicating a graphical program for execution *(see Column 9: 14-28, "A standard WWW "Web" browser 27 such as Netscape [8] is*

initiated on a second computer. To operate the remote control system 1, a user on the second computer 25 specifies the "starter URL" as specified by the coordinated naming convention 5. URLs are defined by the WWW specification and include a named address of a target computer and a filename associated with the target computer.");

- execute the graphical program *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer 19.")*;

- send information describing a user interface of the graphical program over the network to the client computer system after establishing the network connection with the client computer system, wherein the information describing the user interface is useable by the client computer system to display the user interface on the client computer system *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer 19."; Column 9: 31-41, "In response to the starter URL, the server program 21 builds a new file, a GIF image file containing the screen image of the GUI on the first computer, and returns the data of REMOTE.HTM.")*; and

- send information regarding the graphical program over the network to the client computer system after establishing the network connection with the client computer system, wherein the information regarding the graphical program is useable by the client computer system to display the graphical program on the client computer system *(see Column 9: 31-50, "REMOTE.HTM contains appropriate HTML references to the GIF file so that the GIF file will be displayed as a clickable image. In the preferred embodiment, the GIF file thus built is a 256*

color image of the GUI screen of the first computer 19. The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.");

- wherein the user interface is operable to facilitate interaction between a user and the graphical program over the network *(see Column 9: 42-50, "The user in this embodiment will see a screen virtually identical to the GUI screen on the first computer. The user may then click on a menu, button, or other Windows control image.").*

However, Huntsman does not disclose:

- wherein the graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function, and wherein said executing the graphical program comprises executing the block diagram; and

- information regarding the block diagram of the graphical program.

Kodosky discloses:

- wherein a graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function, and wherein executing the graphical program comprises executing the block diagram *(see Column 7: 44-59, "The execution subsystem 24 assigns at least one value to the input variable and executes the execution instructions to produce a value for the output variable. The control processor 26 implements the block diagram editor 22 and the execution subsystem 24 of the preferred embodiment."; Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for*

an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42."); and

- information regarding the block diagram of the graphical program *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to have the graphical program as a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function, and executing the graphical program comprises executing the block diagram; and to include information regarding the block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system.

As per **Claim 60**, the rejection of **Claim 59** is incorporated; and Huntsman further discloses:

- provide information indicating a plurality of graphical programs to the client computer system over the network, wherein the information indicating a plurality of graphical programs is usable by the client computer system to display information indicating the plurality

of graphical programs *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer 19.")*; and

- wherein, in indicating the graphical program for execution, the user input selects the graphical program from the plurality of graphical programs *(see Column 9: 47-50, "The user in this embodiment will see a screen virtually identical to the GUI screen on the first computer. The user may then click on a menu, button, or other Windows control image.")*.

As per **Claim 61**, the rejection of **Claim 59** is incorporated; and Huntsman further discloses:

- receive user input to the graphical program from the client computer system over the network *(see Column 9: 42-50, "The user in this embodiment will see a screen virtually identical to the GUI screen on the first computer. The user may then click on a menu, button, or other Windows control image.")*; and

- provide the user input to the graphical program *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")*;

- wherein the graphical program is operable to respond to the user input *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the*

HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.").

As per **Claim 62**, the rejection of **Claim 59** is incorporated; and Huntsman further discloses:

- wherein the graphical program produces a first output state *(see Column 9: 31-41, "In response to the starter URL, the server program 21 builds a new file, a GIF image file containing the screen image of the GUI on the first computer, and returns the data of REMOTE.HTM.")*; and

- wherein said sending information describing a user interface of the graphical program comprises sending information indicative of the first output state *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.").*

As per **Claim 63**, the rejection of **Claim 62** is incorporated; and Huntsman further discloses:

- wherein the graphical program produces a second output state after the graphical program produces the first output state *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.")*; and

- wherein the program instructions are further executable to send a user interface update indicating the second output state to the client computer system *(see Column 10: 6-10, "Moments later, a user at the second computer 25 will typically select the "REFRESH" hypertext link which sends a reference of REMOTE.HTM of FIG. 10 to the first computer, repeating the behavior just described.").*

As per **Claim 64**, the rejection of **Claim 59** is incorporated; and Huntsman further discloses:

- establish a network connection with each of a plurality of client computer systems *(see Column 8: 11-15, "One or more second computers 25 running a standard, off-the-shelf hypertext browser program 27, can effectuate remote control using the standard, widely installed networking protocols, including those used in the internet [9].")*; and

- send information describing a user interface of the graphical program over the network to each of the plurality of client computer systems after establishing the network

connection with each of the plurality of client computer systems *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.").*

As per **Claim 65**, the rejection of **Claim 64** is incorporated; and Huntsman further discloses:

- send information regarding the graphical program over the network to each of the plurality of client computer systems after establishing the network connection with each of the plurality of client computer systems, wherein the information regarding the graphical program is useable by each of the plurality of client computer systems to display the graphical program *(see Column 9: 31-50, "REMOTE.HTM contains appropriate HTML references to the GIF file so that the GIF file will be displayed as a clickable image. In the preferred embodiment, the GIF file thus built is a 256 color image of the GUI screen of the first computer 19. The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.").*

However, Huntsman does not disclose:

- information regarding the block diagram of the graphical program.

Kodosky discloses:

- information regarding a block diagram of a graphical program *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include information regarding a block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by sending block diagram information between the two computer systems.

As per **Claim 66**, the rejection of **Claim 59** is incorporated; and Huntsman further discloses:

- wherein the graphical program executes to perform a measurement or automation function *(see Column 1: 29-34, "GUI environments tend to use a pointing device, like a mouse, in addition to a keyboard. Instead of typing a textual command, the user of a graphical interface typically selects a button or menu selection with a pointing device such as a mouse and "clicks" on his selection.").*

As per **Claim 67**, the rejection of **Claim 59** is incorporated; and Huntsman further discloses:

- wherein the network is the Internet *(see Column 9: 12-14, "The second computer 25 is connected to the first computer over the internet 31 ...").*

As per **Claim 68**, the rejection of **Claim 59** is incorporated; and Huntsman further discloses:

- wherein the information describing the user interface is useable by the client computer system to display the user interface of the graphical program on a web browser *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.")*; and

- wherein the information regarding the graphical program is useable by the client computer system to display the graphical program on the web browser *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.").*

However, Huntsman does not disclose:

- information regarding the block diagram of the graphical program.

Kodosky discloses:

- information regarding a block diagram of a graphical program *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual*

representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.").

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include information regarding a block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by displaying the block diagram information for a user to control.

As per **Claim 69**, the rejection of **Claim 59** is incorporated; and Huntsman further discloses:

- receive user input specifying an edit to the graphical program from the client computer system over the network *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")*; and

- edit the graphical program according to the user input specifying the edit *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means*

7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.").

However, Huntsman does not disclose:

- an edit to the block diagram of the graphical program.

Kodosky discloses:

- an edit to a block diagram of a graphical program *(see Column 18: 47-51, "FIG. 25 shows the EDIT menu selections ... CLEAR is useful for removing items from the active window, e.g., selected wires and structures from the block diagram window, or controls from the front panel window.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include an edit to a block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by allowing a user to edit the block diagram information from a remote location.

As per **Claim 70**, the rejection of **Claim 59** is incorporated; however, Huntsman does not disclose:

- wherein the user interface of the graphical program comprises at least one input variable icon for providing inputs to the block diagram and at least one output variable icon for displaying outputs produced by the block diagram.

Kodosky discloses:

- wherein a user interface of a graphical program comprises at least one input variable icon for providing inputs to a block diagram and at least one output variable icon for displaying outputs produced by the block diagram *(see Column 8: 13-19, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include wherein the user interface of the graphical program comprises at least one input variable icon for providing inputs to a block diagram and at least one output variable icon for displaying outputs produced by the block diagram. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by allowing a user to configure input and output data for a block diagram via a user interface.

As per **Claim 71**, the rejection of **Claim 59** is incorporated; and Huntsman further discloses:

- receive input of at least one input variable from the client computer system over the network *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")*; and

- providing the output of the at least one output variable to the client computer system over the network for display *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.")*.

However, Huntsman does not disclose:

- the block diagram executing using the input of the at least one input variable; and

- the block diagram generating an output of at least one output variable.

Kodosky discloses:

- a block diagram executing using the input of the at least one input variable *(see Column 13: 47-55, "... execution instructions can be constructed by constructing a visual display in which at least one input variable produces at least output variable according to a displayed procedure.")*; and

- the block diagram generating an output of at least one output variable *(see Column 13: 47-55, "... execution instructions can be constructed by constructing a visual display in which at least one input variable produces at least output variable according to a displayed procedure.")*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include a block diagram executing using the input of the at least one input variable; and the block diagram generating an output of at least one output variable. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by executing the block diagram from a remote location using input and output data for the block diagram configured by a user.

As per **Claim 72**, the rejection of **Claim 59** is incorporated; however, Huntsman does not disclose:

- wherein the graphical program implements a virtual instrument; and
- wherein the user interface of the graphical program comprises a front panel of the virtual instrument.

Kodosky discloses:

- wherein a graphical program implements a virtual instrument *(see Figure 3: 40)*; and
- wherein a user interface of the graphical program comprises a front panel of the virtual instrument *(see Figure 3: 42)*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include wherein the graphical program implements a virtual instrument; and wherein the user interface of the graphical program comprises a front panel of the virtual instrument. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system.

As per **Claim 73**, Huntsman discloses:

- executing the graphical program on the first computer , wherein the first computer and the second computer are connected over a network *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer 19."; Column 9: 12-14, "The second computer 25 is connected to the first computer over the internet 31 ...")*;

- providing information describing the user interface of the graphical program to the second computer during said executing, wherein said providing comprises the first computer providing the information describing the user interface of the graphical program over the network to the second computer, and wherein the information describing the user interface is useable by the second computer to display the user interface of the graphical program on the second computer *(see Column 9: 31-41, "In response to the starter URL, the server program 21 builds a new file, a GIF image file containing the screen image of the GUI on the first computer, and returns the data of REMOTE.HTM.")*; and

- providing information regarding the graphical program to the second computer over the network, wherein said providing comprises the first computer providing the information regarding the graphical program over the network to the second computer, wherein the information regarding the graphical program is useable by the second computer to display the graphical program on the second computer *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer 19."; Column 9: 31-50, "REMOTE.HTM contains appropriate HTML references to the GIF file so that the GIF file will be displayed as a clickable image. In the preferred embodiment, the GIF file thus built is a 256 color image of the GUI screen of the first computer 19. The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.")*;

- wherein the user interface facilitates interaction between a user of the second computer and the graphical program executing on the first computer *(see Column 9: 42-50, "The user in this embodiment will see a screen virtually identical to the GUI screen on the first computer. The user may then click on a menu, button, or other Windows control image.")*.

However, Huntsman does not disclose:

- wherein the graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function, and wherein said executing the graphical program comprises executing the block diagram; and

- information regarding the block diagram of the graphical program.

Kodosky discloses:

- wherein a graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function, and wherein said executing the graphical program comprises executing the block diagram *(see Column 7: 44-59, "The execution subsystem 24 assigns at least one value to the input variable and executes the execution instructions to produce a value for the output variable. The control processor 26 implements the block diagram editor 22 and the execution subsystem 24 of the preferred embodiment."; Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.")*; and

- information regarding the block diagram of the graphical program *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.")*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to have the graphical program as a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function, and wherein said executing the graphical program comprises executing the block diagram; and to include information regarding the block diagram of the graphical program. The modification

would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system.

As per **Claim 74**, the rejection of **Claim 73** is incorporated; and Huntsman further discloses:

- providing information describing the user interface of the graphical program to a plurality of computers over the network during said executing, where the information describing the user interface of the graphical program is useable by each of the plurality of computers to display the user interface of the graphical program *(see Column 8: 11-15, "One or more second computers 25 running a standard, off-the-shelf hypertext browser program 27, can effectuate remote control using the standard, widely installed networking protocols, including those used in the internet [9]."; Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.").*

As per **Claim 75**, the rejection of **Claim 73** is incorporated; and Huntsman further discloses:

- wherein the information describing the user interface is useable by the second computer to display the user interface of the graphical program on a web browser of the second computer *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the*

HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image."); and

- wherein the information regarding the graphical program is useable by the second computer to display the graphical program on the web browser of the second computer *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.").*

However, Huntsman does not disclose:

- information regarding the block diagram of the graphical program.

Kodosky discloses:

- information regarding a block diagram of a graphical program *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include information regarding a block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing

on one computer system from another computer system by displaying the block diagram information for a user to control.

As per **Claim 76**, the rejection of **Claim 73** is incorporated; and Huntsman further discloses:

- the graphical program executing on the first computer responding to user input received to the graphical program via the displayed user interface on the second computer; wherein the user input is provided to the first computer over the network *(see Column 9: 42-57, "The user in this embodiment will see a screen virtually identical to the GUI screen on the first computer. The user may then click on a menu, button, or other Windows control image. The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.").*

As per **Claim 77**, the rejection of **Claim 73** is incorporated; and Huntsman further discloses:

- wherein the graphical program produces a second output state after the graphical program produces a first output state *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename*

as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7."); and

- providing a user interface update over the network indicating the second output state, where the user interface update is useable by the second computer to update the user interface displayed on the second computer *(see Column 10: 6-10, "Moments later, a user at the second computer 25 will typically select the "REFRESH" hypertext link which sends a reference of REMOTE.HTM of FIG. 10 to the first computer, repeating the behavior just described.").*

As per **Claim 78**, the rejection of **Claim 73** is incorporated; and Huntsman further discloses:

- receiving user input specifying an edit to the graphical program to the first computer from the second computer over the network *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")*; and

- editing the graphical program according to the user input specifying the edit, wherein said editing is performed by the first computer *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI*

control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.").

However, Huntsman does not disclose:

- an edit to the block diagram of the graphical program.

Kodosky discloses:

- an edit to a block diagram of a graphical program *(see Column 18: 47-51, "FIG. 25 shows the EDIT menu selections ... CLEAR is useful for removing items from the active window, e.g., selected wires and structures from the block diagram window, or controls from the front panel window.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include an edit to a block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by allowing a user to edit the block diagram information from a remote location.

As per **Claim 79**, the rejection of **Claim 73** is incorporated; and Huntsman further discloses:

- receiving input of at least one input variable to the first computer from the second computer over the network *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")*; and

- providing the output of the at least one output variable to the second computer over the network, wherein the output is displayable on the second computer *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.").*

However, Huntsman does not disclose:

- wherein the user interface of the graphical program comprises at least one input variable icon for providing inputs to the block diagram and at least one output variable icon for displaying outputs produced by the block diagram;

- the block diagram executing using the input of the at least one input variable; and

- the block diagram generating an output of at least one output variable.

Kodosky discloses:

- wherein a user interface of a graphical program comprises at least one input variable icon for providing inputs to a block diagram and at least one output variable icon for displaying outputs produced by the block diagram *(see Column 8: 13-19, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.")*;

- the block diagram executing using the input of the at least one input variable *(see Column 13: 47-55, "... execution instructions can be constructed by constructing a visual display in which at least one input variable produces at least output variable according to a displayed procedure.")*; and

- the block diagram generating an output of at least one output variable *(see Column 13: 47-55, "... execution instructions can be constructed by constructing a visual display in which at least one input variable produces at least output variable according to a displayed procedure.")*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include wherein the user interface of the graphical program comprises at least one input variable icon for providing inputs to a block diagram and at least one output variable icon for displaying outputs produced by the block diagram; the block diagram executing using the input of the at least one input variable; and the block diagram generating an output of at least one output variable. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a

virtual instrument that is executing on one computer system from another computer system by allowing a user to configure input and output data for a block diagram via a user interface and executing the block diagram from a remote location using input and output data for the block diagram configured by the user.

As per **Claim 80**, the rejection of **Claim 73** is incorporated; however, Huntsman does not disclose:

- wherein the graphical program implements a virtual instrument; and
- wherein the user interface of the graphical program comprises a front panel of the virtual instrument.

Kodosky discloses:

- wherein a graphical program implements a virtual instrument *(see Figure 3: 40)*; and
- wherein a user interface of the graphical program comprises a front panel of the virtual instrument *(see Figure 3: 42)*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include wherein the graphical program implements a virtual instrument; and wherein the user interface of the graphical program comprises a front panel of the virtual instrument. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system.

As per **Claim 81**, Huntsman discloses:

- a first computer *(see Figure 4: 19)*, comprising:
  - a processor *(see Figure 4: 19)*; and
  - a memory *(see Figure 4: 19)*, coupled to the processor;
- wherein the first computer is operable to couple to a network *(see Figure 4: 19 and 31)*;
- wherein the memory stores a graphical program *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer 19.")*;
- wherein the first computer is operable to execute the graphical program and provide information describing a user interface of the graphical program over the network to a second computer during said executing *(see Column 9: 31-41, "In response to the starter URL, the server program 21 builds a new file, a GIF image file containing the screen image of the GUI on the first computer, and returns the data of REMOTE.HTM.")*;
- wherein the information describing the user interface over the network is useable by the second computer to display the user interface of the graphical program *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.")*;
- wherein the user interface facilitates interaction between a user of the second computer and the graphical program executing on the first computer *(see Column 9: 42-50, "The*

user in this embodiment will see a screen virtually identical to the GUI screen on the first computer. The user may then click on a menu, button, or other Windows control image."); and

- wherein the first computer is operable to provide information regarding the graphical program over the network to the second computer, wherein the information regarding the graphical program is useable by the second computer to display the graphical program on the second computer *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer 19."; Column 9: 31-50, "REMOTE.HTM contains appropriate HTML references to the GIF file so that the GIF file will be displayed as a clickable image. In the preferred embodiment, the GIF file thus built is a 256 color image of the GUI screen of the first computer 19. The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.").*

However, Huntsman does not disclose:

- wherein the graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function;

- wherein said executing the graphical program comprises executing the block diagram; and

- information regarding the block diagram of the graphical program.

Kodosky discloses:

- wherein a graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.")*;

- wherein said executing the graphical program comprises executing the block diagram *(see Column 7: 44-59, "The execution subsystem 24 assigns at least one value to the input variable and executes the execution instructions to produce a value for the output variable. The control processor 26 implements the block diagram editor 22 and the execution subsystem 24 of the preferred embodiment.")*; and

- information regarding the block diagram of the graphical program *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.")*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to have the graphical program as a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function; wherein said executing the graphical program comprises executing the block diagram; and to include information regarding the block diagram of the graphical program. The modification

would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system.

As per **Claim 82**, Huntsman discloses:

- receive user input at a first computer indicating a graphical program, wherein the graphical program is stored on a server computer *(see Column 9: 14-28, "A standard WWW "Web" browser 27 such as Netscape [8] is initiated on a second computer. To operate the remote control system 1, a user on the second computer 25 specifies the "starter URL" as specified by the coordinated naming convention 5. URLs are defined by the WWW specification and include a named address of a target computer and a filename associated with the target computer.")*;

- provide the user input indicating the graphical program over a network to the server computer *(see Column 9: 14-28, "A standard WWW "Web" browser 27 such as Netscape [8] is initiated on a second computer. To operate the remote control system 1, a user on the second computer 25 specifies the "starter URL" as specified by the coordinated naming convention 5. URLs are defined by the WWW specification and include a named address of a target computer and a filename associated with the target computer.")*;

- receive information describing a user interface of the graphical program from the server computer over the network during execution of the graphical program on the server computer *(see Column 9: 31-41, "In response to the starter URL, the server program 21 builds a new file, a GIF image file containing the screen image of the GUI on the first computer, and returns the data of REMOTE.HTM.")*;

- receive information regarding the graphical program from the server computer over the network *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer 19."; Column 9: 31-41, "REMOTE.HTM contains appropriate HTML references to the GIF file so that the GIF file will be displayed as a clickable image. In the preferred embodiment, the GIF file thus built is a 256 color image of the GUI screen of the first computer 19.")*;

- display the user interface of the graphical program at the first computer based on the information describing a user interface *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.")*; and

- display the graphical program at the first computer based on the information regarding the graphical program *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.")*;

- wherein the user interface is operable to facilitate interaction between a user and the graphical program executing on the server computer *(see Column 9: 42-50, "The user in this embodiment will see a screen virtually identical to the GUI screen on the first computer. The user may then click on a menu, button, or other Windows control image.")*.

However, Huntsman does not disclose:

- wherein the graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function; and

- information regarding the block diagram of the graphical program.

Kodosky discloses:

- wherein a graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function *(see Column 7: 44-59, "The execution subsystem 24 assigns at least one value to the input variable and executes the execution instructions to produce a value for the output variable. The control processor 26 implements the block diagram editor 22 and the execution subsystem 24 of the preferred embodiment."; Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.")*; and

- information regarding the block diagram of the graphical program *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.")*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to have the graphical program as a block diagram that comprises a

plurality of interconnected function icons representing graphical data flow of a desired function; and to include information regarding the block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system.

As per **Claim 83**, the rejection of **Claim 82** is incorporated; and Huntsman further discloses:

- wherein the graphical program executes to perform a measurement or automation function *(see Column 1: 29-34, "GUI environments tend to use a pointing device, like a mouse, in addition to a keyboard. Instead of typing a textual command, the user of a graphical interface typically selects a button or menu selection with a pointing device such as a mouse and "clicks" on his selection.").*

As per **Claim 84**, the rejection of **Claim 82** is incorporated; and Huntsman further discloses:

- establish a network connection with the server computer over the network after said receiving user input indicating the graphical program *(see Column 9: 14-28, "A standard WWW "Web" browser 27 such as Netscape [8] is initiated on a second computer. To operate the remote control system 1, a user on the second computer 25 specifies the "starter URL" as specified by the coordinated naming convention 5. URLs are defined by the WWW specification and include a named address of a target computer and a filename associated with the target computer.");*

- wherein said receiving information describing the user interface and said receiving information regarding the graphical program are performed after said user input indicating the graphical program and after said establishing a network connection *(see Column 9: 14-28, "A standard WWW "Web" browser 27 such as Netscape [8] is initiated on a second computer. To operate the remote control system 1, a user on the second computer 25 specifies the "starter URL" as specified by the coordinated naming convention 5. URLs are defined by the WWW specification and include a named address of a target computer and a filename associated with the target computer.").*

However, Huntsman does not disclose:

- information regarding the block diagram of the graphical program.

Kodosky discloses:

- information regarding a block diagram of a graphical program *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include information regarding a block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by sending block diagram information between the two computer systems.

As per **Claim 85**, the rejection of **Claim 84** is incorporated; and Huntsman further discloses:

- wherein the graphical program is already executing on the server computer when said establishing a network connection occurs *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer 19.").*

As per **Claim 86**, the rejection of **Claim 82** is incorporated; and Huntsman further discloses:

- wherein to display the user interface of the graphical program, the program instructions are executable to display the user interface of the graphical program on a web browser *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.").*

As per **Claim 87**, the rejection of **Claim 82** is incorporated; and Huntsman further discloses:

- receive user input to the graphical program via the displayed user interface *(see Column 9: 42-50, "The user in this embodiment will see a screen virtually identical to the GUI*

screen on the first computer. The user may then click on a menu, button, or other Windows control image."); and

- provide the user input to the server computer over the network for input to the graphical program executing on the server computer *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.").*

As per **Claim 88**, the rejection of **Claim 82** is incorporated; and Huntsman further discloses:

- wherein the graphical program produces a first output state *(see Column 9: 31-41, "In response to the starter URL, the server program 21 builds a new file, a GIF image file containing the screen image of the GUI on the first computer, and returns the data of REMOTE.HTM.")*; and

- wherein said displaying the user interface includes displaying the user interface illustrating the first output state *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.").*

As per **Claim 89**, the rejection of **Claim 82** is incorporated; and Huntsman further discloses:

- wherein the graphical program produces a second output state after the graphical program produces a first output state *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.")*;

- receive a user interface update over the network indicating the second output state *(see Column 10: 6-10, "Moments later, a user at the second computer 25 will typically select the "REFRESH" hypertext link which sends a reference of REMOTE.HTM of FIG. 10 to the first computer, repeating the behavior just described.")*; and

- update the user interface in response to the user interface update *(see Column 10: 6-10, "Moments later, a user at the second computer 25 will typically select the "REFRESH" hypertext link which sends a reference of REMOTE.HTM of FIG. 10 to the first computer, repeating the behavior just described.")*.

As per **Claim 90**, the rejection of **Claim 82** is incorporated; and Huntsman further discloses:

- receive user input specifying an edit to the graphical program *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")*; and

- provide the user input specifying the edit to the server computer over the network *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")*;

- wherein the first computer is operable to edit the graphical program according to the user input specifying the edit *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.")*.

However, Huntsman does not disclose:

- an edit to the block diagram of the graphical program.

Kodosky discloses:

- an edit to a block diagram of a graphical program *(see Column 18: 47-51, "FIG. 25 shows the EDIT menu selections ... CLEAR is useful for removing items from the active window, e.g., selected wires and structures from the block diagram window, or controls from the front panel window.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include an edit to a block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by allowing a user to edit the block diagram information from a remote location.

As per **Claim 91**, the rejection of **Claim 82** is incorporated; and Huntsman further discloses:

- wherein said indicating the graphical program comprises providing a uniform resource locator (URL) *(see Column 9: 14-28, "A standard WWW "Web" browser 27 such as Netscape [8] is initiated on a second computer. To operate the remote control system 1, a user on the second computer 25 specifies the "starter URL" as specified by the coordinated naming convention 5. URLs are defined by the WWW specification and include a named address of a target computer and a filename associated with the target computer.").*

As per **Claim 92**, the rejection of **Claim 82** is incorporated; and Huntsman further discloses:

- display information indicating a plurality of graphical programs on the first computer *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer 19.")*; and

- wherein, in indicating the graphical program on the first computer, the user input selects the graphical program from the plurality of graphical programs *(see Column 9: 47-50, "The user in this embodiment will see a screen virtually identical to the GUI screen on the first computer. The user may then click on a menu, button, or other Windows control image.").*

As per **Claim 93**, the rejection of **Claim 82** is incorporated; however, Huntsman does not disclose:

- wherein the user interface of the graphical program comprises at least one input variable icon for providing inputs to the block diagram and at least one output variable icon for displaying outputs produced by the block diagram.

Kodosky discloses:

- wherein a user interface of a graphical program comprises at least one input variable icon for providing inputs to a block diagram and at least one output variable icon for displaying outputs produced by the block diagram *(see Column 8: 13-19, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure*

by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.").

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include wherein the user interface of the graphical program comprises at least one input variable icon for providing inputs to a block diagram and at least one output variable icon for displaying outputs produced by the block diagram. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by allowing a user to configure input and output data for a block diagram via a user interface.

As per **Claim 94**, the rejection of **Claim 82** is incorporated; and Huntsman further discloses:

- receive user input manipulating input of at least one input variable *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")*;

- provide the user input of the at least one input variable to the server computer over the network *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP*

protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.");

- receive output of at least one output variable from the server computer over the network *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.")*; and

- display the output of the at least one output variable *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.").*

However, Huntsman does not disclose:

- wherein the output is generated by the block diagram executing using the user input of the at least one input variable.

Kodosky discloses:

- wherein output is generated by a block diagram executing using user input of at least one input variable *(see Column 13: 47-55, "... execution instructions can be constructed by constructing a visual display in which at least one input variable produces at least output variable according to a displayed procedure.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include wherein the output is generated by a block diagram executing using the user input of the at least one input variable. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by allowing a user to edit the block diagram information from a remote location.

As per **Claim 95**, the rejection of **Claim 82** is incorporated; however, Huntsman does not disclose:

- wherein the graphical program implements a virtual instrument; and
- wherein the user interface of the graphical program comprises a front panel of the virtual instrument.

Kodosky discloses:

- wherein a graphical program implements a virtual instrument *(see Figure 3: 40)*; and

- wherein a user interface of the graphical program comprises a front panel of the virtual instrument *(see Figure 3: 42)*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include wherein the graphical program implements a virtual instrument; and wherein the user interface of the graphical program comprises a front panel of the virtual instrument. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system.

As per **Claim 96**, Huntsman discloses:

- receiving user input to the second computer, wherein the user input indicates the graphical program on the first computer, wherein the first computer and the second computer are connected over a network *(see Column 9: 12-14, "The second computer 25 is connected to the first computer over the internet 31 ..."; Column 9: 14-28, "A standard WWW "Web" browser 27 such as Netscape [8] is initiated on a second computer. To operate the remote control system 1, a user on the second computer 25 specifies the "starter URL" as specified by the coordinated naming convention 5. URLs are defined by the WWW specification and include a named address of a target computer and a filename associated with the target computer.")*;

- receiving information describing the graphical user interface of the graphical program at the second computer from the first computer over the network during execution of the graphical program on the first computer *(see Column 9: 31-41, "In response to the starter URL,*

the server program 21 builds a new file, a GIF image file containing the screen image of the GUI on the first computer, and returns the data of REMOTE.HTM.");

- receiving information regarding the graphical program at the second computer from the first computer over the network *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer 19."; Column 9: 31-41, "REMOTE.HTM contains appropriate HTML references to the GIF file so that the GIF file will be displayed as a clickable image. In the preferred embodiment, the GIF file thus built is a 256 color image of the GUI screen of the first computer 19.")*;

- displaying the graphical user interface of the graphical program on the second computer based on the information describing the graphical user interface *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.")*; and

- displaying the graphical program on the second computer, using the information regarding the graphical program *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.")*;

- wherein the graphical user interface facilitates interaction between a user of the second computer and the graphical program executing on the first computer *(see Column 9: 42-*

50, "The user in this embodiment will see a screen virtually identical to the GUI screen on the first computer. The user may then click on a menu, button, or other Windows control image.").

However, Huntsman does not disclose:

- wherein the graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function; and

- information regarding the block diagram of the graphical program.

Kodosky discloses:

- wherein a graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function *(see Column 7: 44-59, "The execution subsystem 24 assigns at least one value to the input variable and executes the execution instructions to produce a value for the output variable. The control processor 26 implements the block diagram editor 22 and the execution subsystem 24 of the preferred embodiment."; Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.")*; and

- information regarding the block diagram of the graphical program *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to have the graphical program as a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function; and to include information regarding the block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system.

As per **Claim 97**, the rejection of **Claim 96** is incorporated; and Huntsman further discloses:

- establish a network connection with the first computer over the network after said receiving user input indicating the graphical program *(see Column 9: 14-28, "A standard WWW "Web" browser 27 such as Netscape [8] is initiated on a second computer. To operate the remote control system 1, a user on the second computer 25 specifies the "starter URL" as specified by the coordinated naming convention 5. URLs are defined by the WWW specification and include a named address of a target computer and a filename associated with the target computer.")*;

- wherein said receiving information describing the graphical user interface and said receiving information regarding the graphical program are performed after said user input indicating the graphical program and after said establishing a network connection *(see Column 9: 14-28, "A standard WWW "Web" browser 27 such as Netscape [8] is initiated on a second computer. To operate the remote control system 1, a user on the second computer 25 specifies*

the "starter URL" as specified by the coordinated naming convention 5. URLs are defined by the WWW specification and include a named address of a target computer and a filename associated with the target computer.").

However, Huntsman does not disclose:

- information regarding the block diagram of the graphical program.

Kodosky discloses:

- information regarding a block diagram of a graphical program *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include information regarding a block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by sending block diagram information between the two computer systems.

As per **Claim 98**, the rejection of **Claim 96** is incorporated; and Huntsman further discloses:

- wherein displaying the graphical user interface of the graphical program comprises displaying the graphical user interface of the graphical program on a web browser *(see Column*

9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.").

As per **Claim 99**, the rejection of **Claim 96** is incorporated; and Huntsman further discloses:

- receiving user input to the graphical program via the displayed graphical user interface *(see Column 9: 42-50, "The user in this embodiment will see a screen virtually identical to the GUI screen on the first computer. The user may then click on a menu, button, or other Windows control image.")*; and

- providing the user input to the first computer over the network for input to the graphical program executing on the first computer *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.").*

As per **Claim 100**, the rejection of **Claim 96** is incorporated; and Huntsman further discloses:

- receiving user input specifying an edit to the graphical program *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")*; and

- providing the user input specifying the edit to the first computer over the network *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")*;

- wherein the user input specifying the edit is useable by the first computer to edit the graphical program *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.")*.

However, Huntsman does not disclose:

- an edit to the block diagram of the graphical program.

Kodosky discloses:

- an edit to a block diagram of a graphical program *(see Column 18: 47-51, "FIG. 25 shows the EDIT menu selections ... CLEAR is useful for removing items from the active window, e.g., selected wires and structures from the block diagram window, or controls from the front panel window.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include an edit to a block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by allowing a user to edit the block diagram information from a remote location.

As per **Claim 101**, the rejection of **Claim 96** is incorporated; however, Huntsman does not disclose:

- wherein the graphical user interface of the graphical program comprises at least one input variable icon for providing inputs to the block diagram and at least one output variable icon for displaying outputs produced by the block diagram.

Kodosky discloses:

- wherein a graphical user interface of a graphical program comprises at least one input variable icon for providing inputs to a block diagram and at least one output variable icon for displaying outputs produced by the block diagram *(see Column 8: 13-19, "The virtual instrument*

40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.").

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include wherein the graphical user interface of the graphical program comprises at least one input variable icon for providing inputs to a block diagram and at least one output variable icon for displaying outputs produced by the block diagram. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by allowing a user to configure input and output data for a block diagram via a user interface.

As per **Claim 102**, the rejection of **Claim 96** is incorporated; and Huntsman further discloses:

- receiving user input manipulating input of at least one input variable *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")*;

- providing the user input of the at least one input variable to the first computer over the network *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")*;

- receiving output of at least one output variable from the first computer *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.")*; and

- displaying the output of at least one output variable on the second computer *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7.")*.

However, Huntsman does not disclose:

- wherein the output is generated by the block diagram executing using the user input of the at least one input variable.

Kodosky discloses:

- wherein output is generated by a block diagram executing using user input of at least one input variable *(see Column 13: 47-55, "... execution instructions can be constructed by constructing a visual display in which at least one input variable produces at least output variable according to a displayed procedure.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include wherein the output is generated by a block diagram executing using the user input of the at least one input variable. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by allowing a user to edit the block diagram information from a remote location.

As per **Claim 103**, the rejection of **Claim 96** is incorporated; however, Huntsman does not disclose:

- wherein the graphical program implements a virtual instrument; and
- wherein the user interface of the graphical program comprises a front panel of the virtual instrument.

Kodosky discloses:

- wherein a graphical program implements a virtual instrument *(see Figure 3: 40)*; and
- wherein a user interface of the graphical program comprises a front panel of the virtual instrument *(see Figure 3: 42)*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to include wherein the graphical program implements a virtual instrument; and wherein the user interface of the graphical program comprises a front panel of the virtual instrument. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system.

As per **Claim 104**, Huntsman discloses:

- a processor *(see Figure 4: 25)*; and
- a memory *(see Figure 4: 25)*, coupled to the processor;
- a network port operable to couple to a network *(see Figure 4: 25 and 31)*;
- wherein the memory stores program instructions executable by the processor to:
  - receive user input specifying a graphical program *(see Column 9: 14-28, "A standard WWW "Web" browser 27 such as Netscape [8] is initiated on a second computer. To operate the remote control system 1, a user on the second computer 25 specifies the "starter URL" as specified by the coordinated naming convention 5. URLs are defined by the WWW specification and include a named address of a target computer and a filename associated with the target computer.")*;

- provide the user input specifying the graphical program over a network to a server computer *(see Column 9: 14-28, "A standard WWW "Web" browser 27 such as Netscape [8] is initiated on a second computer. To operate the remote control system 1, a user on the second computer 25 specifies the "starter URL" as specified by the coordinated naming convention 5. URLs are defined by the WWW specification and include a named address of a target computer and a filename associated with the target computer.")*;

- receive information describing a user interface of the graphical program from the server computer over the network during execution of the graphical program on the server computer *(see Column 9: 31-41, "In response to the starter URL, the server program 21 builds a new file, a GIF image file containing the screen image of the GUI on the first computer, and returns the data of REMOTE.HTM.")*;

- receive information regarding the graphical program from the server computer over the network *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer 19."; Column 9: 31-41, "REMOTE.HTM contains appropriate HTML references to the GIF file so that the GIF file will be displayed as a clickable image. In the preferred embodiment, the GIF file thus built is a 256 color image of the GUI screen of the first computer 19.")*;

- display the user interface of the graphical program based on the information describing a user interface *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and*

retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image."); and

- display the graphical program based on the information regarding the graphical program *(see Column 9: 42-50, "The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.")*;

- wherein the user interface facilitates interaction between a user and the graphical program executing on the server computer *(see Column 9: 42-50, "The user in this embodiment will see a screen virtually identical to the GUI screen on the first computer. The user may then click on a menu, button, or other Windows control image.").*

However, Huntsman does not disclose:

- wherein the graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function; and

- information regarding the block diagram of the graphical program.

Kodosky discloses:

- wherein a graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function *(see Column 7: 44-59, "The execution subsystem 24 assigns at least one value to the input variable and executes the execution instructions to produce a value for the output variable. The control processor 26 implements the block diagram editor 22 and the execution subsystem 24 of the preferred embodiment."; Column 8: 8-23, "The virtual instrument 40 also includes a block*

diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42."); and

- information regarding the block diagram of the graphical program *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.")*.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Kodosky into the teaching of Huntsman to modify Huntsman's invention to have the graphical program as a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function; and to include information regarding the block diagram of the graphical program. The modification would be obvious because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system.

(10) Response to Argument

Claims 59, 61-67, 70-73, 76, 77, 79-89, 91-99, and 101-104:

In the Appeal Brief, Appellant argues:

a) It is improper for the Examiner to ignore the definition of "block diagram" contained in the claim ("a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function"), in an attempt to blur the distinctions between text-based programs, such as Huntsman's MS-Windows programs, and Appellant's graphical programs. Applicant respectfully submits that one of skill in the programming arts readily understands that text-based programs are written in text-based programming languages, e.g., FORTRAN, PASCAL, C, C++, etc., i.e., have textual source code, and may or may not include GUIs, whereas graphical programs have graphical source code, i.e., a block diagram comprising interconnected graphical program icons or nodes. For example, the cited MS-Windows and Windows Explorer are textual programs, even though they implement GUIs, and the cited folder and file icons are GUI elements, not graphical source code of the programs, and certainly not interconnected icons of a block diagram of a graphical program as recited in claim 59. In other words, the fact that a program's GUI includes "icons, panels, and windows" does not make the program a "graphical program" as defined in the present claim. Thus, these cited icons and other GUI elements are not germane to these claimed features.

(See Appeal Brief – page 8 to page 9.)

Examiner's response:

a) Examiner disagrees. Appellant's arguments are not persuasive for at least the following reasons:

First, without acquiescing to the Appellant's assertion that the fact that a program's GUI includes "icons, panels, and windows" does not make the program a "graphical program," the

Examiner first submits that Huntsman's MS-Windows program is one type of graphical program, albeit not a graphical program which includes a block diagram as claimed. However, Kodosky is relied upon by the Examiner for its teaching of a graphical program which includes a block diagram. Thus, as the claims are interpreted as broadly as their terms reasonably allow (see MPEP § 2111.01(I)), the interpretation of a broad limitation of a "graphical program" as a MS-Windows program and the like by one of ordinary skill in the art is considered to be reasonable by its plain meaning.

Second, it is noted that the claims do not require graphical source code of a program. Instead, the claims recite, exactly, "graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function." Under the broadest reasonable interpretation, the claimed feature of "a plurality of interconnected function icons" can be reasonably interpreted by one of ordinary skill in the art as graphical icons of the program and thus, the claims are not limited to the scope of graphical source code of the program. Appellant is reminded that in order for such limitations to be considered, the claim language requires to specifically recite such limitations in the claims, otherwise broadest reasonable interpretations of the broadly claimed limitations are deemed to be proper.

Third, with respect to the Appellant's assertion that the fact that a program's GUI includes "icons, panels, and windows" does not make the program a "graphical program," the Examiner respectfully submits that Huntsman clearly discloses a graphical program *(see Column 8: 20-23, "The executing GUI program 23 can be any MS-Windows program including the program manager, and is generally whatever program is in the foreground of the first computer*

19."). Note that Huntsman discloses that the GUI-based program can be any MS-Windows program. Thus, one of ordinary skill in the art would readily comprehend that a MS-Windows program is a graphical program (*e.g.*, icons, panels, and windows) that is manipulated by a user using a GUI (*e.g.*, buttons and menus). An example of a MS-Windows program is Windows Explorer. As the Appellant is well aware, Windows Explorer is a file manager program that provides a GUI for accessing the file systems. The file systems are represented by various folder icons and file icons in a window view. Therefore, as can be seen, a MS-Windows program, such as Windows Explorer, is a graphical program with a GUI.

Fourth, with respect to the Appellant's assertion that Huntsman's MS-Windows programs are text-based programs, the Examiner respectfully submits that Huntsman clearly discloses that MS-Windows programs are graphical programs *(see Column 1: 26-36, "With the introduction of the Apple Macintosh, XWindows, and Microsoft Windows, graphical user interfaces (GUIs) have become popular. GUIs are computer user interfaces that are pictorial rather than text based.")*. Note that MS-Windows programs, such as Windows Explorer, are pictorial rather than text-based.

Therefore, for at least the reasons set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

In the Appeal Brief, Appellant argues:

b) The Advisory Action further argues that "the GIF file is a color image of the executing GUI based program", and that "the GIF file represents the GUI features and the graphical program features of the executing GUI-based program". This is incorrect. The GIF file is

directed at the GUI of a text-based program, not the program (source code) itself. As the first paragraph of the Detailed Description makes clear, Huntsman's remote control system has two main components: a GUI-screen-to-hypertext convertor, and a hypertext-to-GUI response means, both directed to translating a textual program's GUI to or from hypertext, e.g., translating a Windows (text-based) program's GUI screen to HTML and GIF data files, and vice versa, respectively, and thus, the GIF file only represents a text-based program's GUI, and is not germane to the claimed graphical program block diagram.

(See Appeal Brief – page 9.)

Examiner's response:

b) Examiner disagrees. With respect to the Appellant's assertion that Huntsman's GIF file only represents a text-based program's GUI, the Examiner respectfully submits that Huntsman's GIF file clearly represents information describing the graphical program *(see Column 9: 31-50, "REMOTE.HTM contains appropriate HTML references to the GIF file so that the GIF file will be displayed as a clickable image. In the preferred embodiment, the GIF file thus built is a 256 color image of the GUI screen of the first computer 19. The browser 27 on the second computer 25 will decode the HTML document file, and locate the references to the GIF file, request and retrieve the GIF file containing the screen image in a separate HTTP request, and display the GIF image on the screen of the second computer 25, as an HTML "clickable" image.").* Note that Huntsman's GIF file is a color image of the executing GUI-based program and displayed as a clickable image. Thus, one of ordinary skill in the art would readily comprehend that the GIF file represents the GUI features and the graphical program features of the executing GUI-based

program. Again, using the example of Windows Explorer, the GIF file of the executing Windows Explorer would contain the GUI features (*e.g.*, buttons and menus) and the graphical program features (*e.g.*, icons, panels, and windows) of the executing Windows Explorer to allow a user remote control of it from a dissimilar computer.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

In the Appeal Brief, Appellant argues:

c) Per the citation, Huntsman's GIF file, as referenced by the REMOTE.HTM, presents the screen image of the client (first) computer, and REMOTE.HTM contains appropriate HTML references to the GIF file so that the GIF file will be displayed as a clickable image, and thus appears to be a GUI. As one of skill in the programming arts would readily understand, such a GUI is not a graphical program, nor, more specifically, a block diagram, as clearly defined in claim 59, where the graphical program includes the block diagram, and where the block diagram includes a plurality of interconnected function icons representing graphical data flow of a desired function, and where executing the graphical program includes executing the block diagram. Nor does Huntsman ever even mention or hint at a graphical program as clearly defined in the claim, and specification.

(See Appeal Brief – page 10.)

Examiner's response:

c) Examiner disagrees. With respect to the Appellant's assertion that Huntsman's GIF file is not a graphical program as defined in the claims, the Examiner respectfully submits that Kodosky is relied upon by the Examiner for its teaching of a graphical program which includes "a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function" *(see Column 8: 8-23, "The virtual instrument 40 also includes a block diagram 46 which graphically provides a visual representation of a procedure by which a specified value for an input variable displayed in the front panel 42 can produce a corresponding value for an output variable in the front panel 42.")* and "wherein executing the graphical program comprises executing the block diagram" *(see Column 7: 44-59, "The execution subsystem 24 assigns at least one value to the input variable and executes the execution instructions to produce a value for the output variable. The control processor 26 implements the block diagram editor 22 and the execution subsystem 24 of the preferred embodiment.")*. Thus, the Appellant's argument regarding Huntsman's GIF file is not a graphical program as defined in the claims is, at best, moot. Therefore, in view of the teaching of Kodosky and the teaching of Huntsman and the state of the art, one of ordinary skill in the art would be motivated to modify Huntsman's MS-Windows program (represented by the GIF file) as a block diagram of a graphical program of a virtual instrument in order to allow the block diagram of the graphical program of the virtual instrument to be remotely controlled by a user so that the user can have access to the block diagram information executing on another computer in a different part of the world *(see Huntsman – Column 1: 10-20)*. Note that those of ordinary skill in the art would readily comprehend that the modification of Huntsman's MS-Windows program as a

block diagram merely involves replacing one type of graphical program with another type of graphical program.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

In the Appeal Brief, Appellant argues:

d) Appellant submits that Huntsman's sending of GUI information and Kodosky's graphical program in combination do not produce this limitation of claim 59. More specifically, if one were to combine Kodosky and Huntsman, the result would NOT involve sending graphical source code over a network to a remote computer. This is because neither reference involves transmitting source code to a remote computer. Huntsman only transmits a graphical user interface, and does not send any of its textual source code. Kodosky nowhere teaches transmission of its graphical source code (block diagram) over a network. Thus the combination lacks a key element of claim 59, and hence is improper.

(See Appeal Brief – page 11, emphasis in original.)

Examiner's response:

d) Examiner disagrees. Appellant's arguments are not persuasive for at least the following reasons:

First, without acquiescing to the Appellant's assertion that neither reference involves transmitting source code to a remote computer, the Examiner first submits that the claims do not require transmitting source code to a remote computer. Instead, the claims recite, exactly,

"send[ing] information regarding the block diagram of the graphical program over the network to the client computer system." The claimed feature of a "block diagram" is defined as comprising "a plurality of interconnected function icons" in the claims. Under the broadest reasonable interpretation, "a plurality of interconnected function icons" can be reasonably interpreted by one of ordinary skill in the art as graphical icons of the program and thus, the claims are not limited to the scope of graphical source code of the program. Appellant is reminded that in order for such limitations to be considered, the claim language requires to specifically recite such limitations in the claims, otherwise broadest reasonable interpretations of the broadly claimed limitations are deemed to be proper.

Second, the Examiner respectfully submits that the combination of Huntsman and Kodosky clearly discloses "send[ing] information regarding the block diagram of the graphical program over the network to the client computer system." In the Office actions, the Examiner acknowledges that Huntsman does not disclose a graphical program including a block diagram as defined in the claims. Instead, Huntsman only discloses a Microsoft Windows® or other GUI-based programs running on a first computer to be remotely controlled by a second computer *(see Abstract; Column 8: 20-23)*. Huntsman also discloses that GUI-based environments (*i.e.*, GUI-based operating systems and GUI-based programs) are pictorial rather than text based *(see Column 1: 26-36)*. Huntsman's invention addresses the problem of controlling GUI-based programs from a dissimilar computer *(see Column 2: 56-67 to Column 3: 1-3)*. Note that, in column 8, lines 20-23 and column 9, lines 31-50, Huntsman discloses that the GUI-based program can be any MS-Windows program. Thus, one of ordinary skill in the art would readily comprehend that a MS-Windows program is a graphical program (*e.g.*, icons, panels, and

windows) that is manipulated by a user using a GUI (*e.g.*, buttons and menus). An example of a MS-Windows program is Windows Explorer. As discussed in the Examiner's response (a) hereinabove, a MS-Windows program, such as Windows Explorer, is a graphical program with a GUI. Huntsman further discloses that a GIF file is a color image of the executing GUI-based program and displayed as a clickable image *(see Column 9: 31-50)*. Thus, one of ordinary skill in the art would also readily comprehend that the GIF file represents the GUI features and the graphical program features of the executing GUI-based program. Hence, contrary to the Appellant's contention, Huntsman clearly discloses sending both GUI information and graphical information of a GUI-based program to a client computer. In the Office actions, the Examiner relied upon Kodosky for its specific teaching of information regarding a block diagram of a graphical program. Thus, in view of the teaching of Kodosky and the teaching of Huntsman and the state of the art, one of ordinary skill in the art would be motivated to modify Huntsman's MS-Windows program (represented by the GIF file) as a block diagram of a graphical program of a virtual instrument in order to allow the block diagram of the graphical program of the virtual instrument to be remotely controlled by a user so that the user can have access to the block diagram information executing on another computer in a different part of the world *(see Huntsman - Column 1: 10-20)*.

Therefore, for at least the reasons set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

In the Appeal Brief, Appellant argues:

e) Moreover, Appellant respectfully submits that one of skill in the art would not combine Huntsman and Kodosky in the manner proposed by the Examiner. As noted above, the Huntsman reference does not teach sending textual code across a network for display to a user. Appellant notes that there is no reason a client user of Huntsman's system would benefit from seeing pages and pages of textual source code. Even an expert programmer would likely to have significant difficulty understanding program functionality based on voluminous text-based program source code. Thus a typical program user would have no use for such textual source code. Thus, Huntsman actually teaches away from claim 1 [sic].

In direct contrast, graphical source code, i.e., Applicant's claimed block diagram, graphically indicates the functionality of the graphical program, and thus there is benefit to sending the block diagram over a network to a client system for viewing by the user.

Appellant again notes that combining Huntsman and Kodosky would simply NOT produce this functionality, at least because Huntsman teaches sending GUI information, not source code (which, being textual, is not useful to typical users), and Kodosky doesn't teach sending either GUI information or the block diagram.

Thus, the alleged combination of Huntsman and Kodosky is invalid, and does not teach these claimed features.

(See Appeal Brief – page 11 to page 12.)

Examiner's response:

e) Examiner disagrees. Appellant's arguments are not persuasive for at least the following reasons:

First, with respect to the Appellant's assertion that Huntsman teaches away from Claim 59, the Examiner respectfully submits that Huntsman's invention is directed to a Microsoft Windows® or other GUI-based programs running on a first computer to be remotely controlled by a second computer by sending a GIF file representation of an executing GUI-based program *(see Abstract; Column 8: 20-23; Column 9: 31-50)*. Examiner agrees with the Appellant's assertion that Huntsman does not teach sending textual code across a network for display to a user because Huntsman's invention relates to graphical programs (*e.g.*, MS-Windows programs). Huntsman does not intend on sending textual code of a program across a network to be remotely controlled. The claimed invention only requires sending a user interface and graphical features of a graphical program. Thus, Huntsman does not teach away from Claim 59 as averred by the Appellant.

Second, with respect to the Appellant's assertion that the alleged combination of Huntsman and Kodosky is invalid, the Examiner respectfully submits that Huntsman is within the field of the Appellant's endeavor and hence is analogous prior art because Huntsman's invention is directed to a remote control system for remotely controlling a Microsoft Windows® or other GUI-based first computer from a second computer over the Internet using only a standard world-wide-web hypertext browser on the second computer. Kodosky is concerned with the same problem which the Appellant sought to be solved and hence is analogous prior art because Kodosky's invention is directed to a method for programming a computer system to control at least one of a virtual instrument and an instrument. Therefore, it is permissible to combine the teaching of Kodosky into the teaching of Huntsman to include the limitations

disclosed by Kodosky since knowledge generally available to one of ordinary skill in the art provides a reason for combining the elements in the manner claimed. See MPEP § 2141.01(a).

Therefore, for at least the reasons set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

In the Appeal Brief, Appellant argues:

f) In the Response to Arguments of the Office Action, the Examiner argues that "one or ordinary skill in the art would readily comprehend that a MS-Windows program is a graphical program (e.g., panels and windows) that is manipulated by a user via a GUI (e.g., buttons and menus)." Again, this is incorrect and improper, since the cited MS-Windows programs do not meet the definition of graphical program as clearly stated in both the claims and the Specification (see quotes above).

(See Appeal Brief – page 12.)

Examiner's response:

f) Examiner disagrees. With respect to the Appellant's assertion that the cited MS-Windows programs do not meet the definition of graphical program as clearly stated in both the claims and the Specification, the Examiner respectfully submits that the Examiner has addressed the Appellant's argument in the Examiner's responses (a) and (c) hereinabove.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

In the Appeal Brief, Appellant argues:

g) In the Response to Arguments, the Examiner further argues that Kodosky is only relied on for "information regarding the block diagram of the graphical program", and that "Huntsman clearly discloses sending information regarding the graphical program over a network to a client system for display…". Appellant respectfully notes that if Kodosky is relied on only for "information regarding the block diagram of the graphical program", that since Huntsman only discloses sending GUI information for a textual program over a network, then the combination does not, and cannot, disclose sending information describing a user interface for a graphical program over a network, *and* sending information describing a block diagram for the graphical program over the network, as claimed. Moreover, as explained repeatedly, Huntsman's programs are not graphical programs, and Huntsman's GUIs are not user interfaces of graphical programs. Appellant respectfully requests that the Examiner give proper weight and consideration to the terms in Appellant's claims, and submits that if such consideration is given, it will be clear that Kodosky and Huntsman do not, and cannot, teach these claimed features.

(See Appeal Brief – page 12 to page 13, emphasis in original.)

Examiner's response:

g) Examiner disagrees. With respect to the Appellant's assertion that the cited MS-Windows programs do not meet the definition of graphical program as clearly stated in both the claims and the Specification, the Examiner respectfully submits that the Examiner has addressed the Appellant's argument in the Examiner's responses (a), (b), and (c) hereinabove.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

In the Appeal Brief, Appellant argues:

h) In the Response to Arguments, the Examiner asserts that Appellant's arguments fail to comply with 37 CFR 1.111(b) because "they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the references. Appellant respectfully disagrees, noting that Appellant has simply explained how the Examiner has failed to show that the cited art discloses the subject matter of Appellant's claims, and has further explained that the references do not and cannot teach key limitations of the claims.

(See Appeal Brief – page 13.)

Examiner's response:

h) Examiner disagrees. With respect to the Appellant's assertion that the Appellant has further explained that the references cannot teach key limitations of the claims, the Examiner respectfully submits that on pages 16-18 of the "Remarks" (received on 10/20/2008), the Appellant has merely provided broad statements characterizing the prior art as failing to teach or suggest certain features of the claimed invention. Thus, the Appellant has failed to present any argument against the Examiner's interpretation of the claims and specifically point out the details in Huntsman and Kodosky that support the Appellant's assertion that Huntsman and Kodosky fail to teach or suggest certain features of the claimed invention. Mere statements concluding that

the prior art does not disclose or suggest a particular claim limitation without any supporting argument or evidence is a general allegation.

In the Appeal Brief, Appellant argues:

i) Regarding the Examiners assertion that Appellant has improperly only attacked the references individually, Appellant respectfully disagrees. Appellant has rebutted specific assertions by the Examiner regarding what each individual references teaches, particularly Huntsman, and respectfully notes that the Examiner's combination argument relies on these assertions, and so fails when the assertions are incorrect. Appellant has also addressed the fact that in combination, these references also fail to produce Appellant's claimed embodiments.

(See Appeal Brief – page 13.)

Examiner's response:

i) Examiner disagrees. With respect to the Appellant's assertion that the Appellant has addressed the fact that in combination, the references fail to produce the Appellant's claimed embodiments, the Examiner respectfully submits that on pages 16-18 of the "Remarks" (received on 10/20/2008), the Appellant has failed to take into consideration Huntsman in view of Kodosky, but instead, merely argued the shortcomings of both Huntsman and Kodosky in a vacuum. Thus, the Appellant has improperly attacked the references individually where the rejections are based on combinations of references.

In the Appeal Brief, Appellant argues:

j) The Office Action further asserts that Huntsman's Abstract teaches a graphical program as recited in the independent claims. The cited Abstract reads thusly:

> A remote control system for remotely controlling a Microsoft Windows or other GUI-based first computer from a second computer over the internet using only a standard world-wide-web hypertext browser on the second computer. The second controlling computer may be dissimilar from the first controlled computer user interface, underlying operating system, and hardware architecture.

As may be seen, nowhere does this citation mention, or even hint at, a graphical program as defined in the claims, specifically, "wherein the graphical program includes a block diagram that comprises a plurality of interconnected function icons representing graphical data flow of a desired function". Thus, the Office Action's assertion is incorrect.

(See Appeal Brief – page 13 to page 14.)

Examiner's response:

j) Examiner disagrees. With respect to the Appellant's assertion that Huntsman's abstract does not mention, or even hint at, a graphical program as defined in the claims, the Examiner respectfully submits that Huntsman's abstract clearly discloses a graphical program. In the abstract, Huntsman discloses a Microsoft Windows® or other GUI-based first computer. Thus, those of ordinary skill in the art would readily recognize that Microsoft Windows® or other GUI-based programs are graphical programs. In the Office actions, the Examiner did not assert that Huntsman discloses a graphical program as defined in the claims (*i.e.*, a block diagram) as averred by the Appellant.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

In the Appeal Brief, Appellant argues:

k) The Examiner continues to argue that Huntsman and Kodosky in combination disclose sending information regarding a block diagram over a network to a client system. However, as explained above at length, this is not the case. Huntsman discloses sending information regarding a Windows computer screen, e.g., a GUI, over a network, and Kodosky discloses a block diagram, but neither reference teaches, or even hints at the desirability of sending information regarding a block diagram over a network. Thus, even in combination, these references would not produce this claimed feature.

(See Appeal Brief – page 14.)

Examiner's response:

k) Examiner disagrees. With respect to the Appellant's assertion that neither reference teaches, or even hints at the desirability of sending information regarding a block diagram over a network, the Examiner respectfully submits that the Examiner has addressed the Appellant's argument in the Examiner's response (c) hereinabove.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

In the Appeal Brief, Appellant argues:

l) Appellant respectfully submits that Appellant has properly addressed the Examiner's assertions, explaining carefully and at length that Huntsman's MS-Windows programs *are not*

graphical programs as defined in the claims for at least the reason that they do not include a block diagram that includes a plurality of interconnected function icons representing graphical data flow of a desired function, and that executing the graphical program includes executing the block diagram. The Examiner simply seems to be ignoring or dismissing these technical features. As another example, Huntsman never even mentions the term "edit", nor describes editing any programs at all, particularly remotely, nor, even more particularly, editing a graphical program, yet the Examiner continues to argue that Huntsman teaches editing a graphical program over a network.

(See Appeal Brief – page 14, emphasis in original.)

Examiner's response:

l) Examiner disagrees. Appellant's arguments are not persuasive for at least the following reasons:

First, with respect to the Appellant's assertion that Huntsman's MS-Windows programs are not graphical programs as defined in the claims, the Examiner respectfully submits that the Examiner has addressed the Appellant's argument in the Examiner's responses (a) and (c) hereinabove.

Second, with respect to the Appellant's assertion that Huntsman never even mentions the term "edit," nor describes editing any programs at all, particularly remotely, nor, even more particularly, editing a graphical program, the Examiner respectfully submits that Huntsman clearly discloses editing a graphical program *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the*

mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click."). Note that the coordinates pointed to by the mouse indicate a region of the GUI-based program (*e.g.*, an icon) that the user is interested in making a change to the GUI-based program (editing the graphical program) in order to control it.

Therefore, for at least the reasons set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

In the Appeal Brief, Appellant argues:

m) As explained above, per this citation, Huntsman's GIF file, as referenced by the REMOTE.HTM, presents the screen image of the client (first) computer, and REMOTE.HTM contains appropriate HTML references to the GIF file so that the GIF file will be displayed as a clickable image, and thus appears to be a GUI. As one of skill in the programming arts would readily understand, such a GUI is not itself a graphical program as defined in the claim, nor, more specifically, is Huntsman's GUI a user interface *for a graphical program.* As defined in claim 59, a graphical program includes the block diagram that includes a plurality of interconnected function icons representing graphical data flow of a desired function, and where executing the graphical program includes executing the block diagram.

As known by those of skill in the art, a textual program that provides graphics functionality is not a graphical program. Moreover, Huntsman never even mentions or hints at a graphical program. Appellant has made this point clearly in previous Responses, yet the

Examiner continues to refer to Huntsman's program as a graphical program. Appellant respectfully submits that it is improper for the Examiner to attempt to redefine key terms in Appellant's claims in a manner contradictory to the definitions in the claims, and in contradiction to the Specification.

(See Appeal Brief – page 15, emphasis in original.)

Examiner's response:

m) Examiner disagrees. With respect to the Appellant's assertion that Huntsman's GUI is not a user interface for a graphical program, the Examiner respectfully submits that the Examiner has addressed the Appellant's argument in the Examiner's responses (a) and (c) hereinabove.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

In the Appeal Brief, Appellant argues:

n) Nor has a proper motivation to combine been provided. The suggested motivation: "because one of ordinary skill in the art would be motivated to remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system", simply states a presumed benefit of Appellant's invention as recited in claim 59, and appears to rely on hindsight analysis using claim 59 as a blueprint, which is improper.

(See Appeal Brief – page 15.)

Examiner's response:

n) Examiner disagrees. Appellant's arguments are not persuasive for at least the following reasons:

First, with respect to the Appellant's assertion that the suggested motivation simply states a presumed benefit of the Appellant's invention, the Examiner respectfully submits that the Examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In the instant case, the motivation to combine Huntsman with Kodosky is found in the knowledge generally available to one of ordinary skill in the art as discussed in the Examiner's responses (c) and (e) hereinabove. Furthermore, there is no requirement that an "express, written motivation to combine must appear in prior art references before a finding of obviousness." See *Ruiz v. A.B. Chance Co.*, 357 F.3d 1270, 1276, 69 USPQ2d 1686, 1690 (Fed. Cir. 2004). See MPEP § 2145 (X)(A).

Second, with respect to the Appellant's assertion that the Examiner appears to rely on hindsight analysis using Claim 59 as a blueprint, the Examiner respectfully submits that it must be recognized that any judgment on obviousness is in a sense necessarily a reconstruction based upon hindsight reasoning. But so long as it takes into account only knowledge which was within the level of ordinary skill at the time the claimed invention was made, and does not include knowledge gleaned only from the Appellant's disclosure, such a reconstruction is proper. See *In re McLaughlin*, 443 F.2d 1392, 170 USPQ 209 (CCPA 1971). Examiner furthers submits that the

concept of remotely controlling a computer or software process is well-known to one of ordinary skill in the computing art and also conventional in the area of software programming. By way of an example and not of limitation, the various patent and non-patent literatures cited by the Examiner are all related to, one form or another, the concept of remotely controlling a computer or software process.

Therefore, for at least the reasons set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

In the Appeal Brief, Appellant argues:

o) The Advisory Action's assertion that it would be obvious "to modify Huntsman's MS-Windows program as a block diagram of a graphical program of a virtual instrument in order to allow the block diagram of the graphical program of the virtual instrument to be remotely controlled by a user so that the user can have access to the block diagram information executing on another computer", is not a proper motivation to combine. First, note that "the block diagram information" of claim 59 is sent to the client (remote) system to display the block diagram, and does not execute on the local computer, contrary to the Advisory Action. Nor is Huntsman's system for remote control of textual-based programs changed substantially by modifying the programs to be graphical programs, since in Huntsman's system (even in combination with Kodosky) it is only the GUI information that is transmitted. The Examiner has attempted to modify Huntsman by simply adding the novel feature of claim 59 "send information regarding the block diagram of the graphical program over the network to the client computer system after establishing the network connection with the client computer system, wherein the information

regarding the block diagram is useable by the client computer system to display the block diagram on a client computer system"—which is neither taught nor suggested by Huntsman or Kodosky, which is improper. Moreover, neither reference even hints at the desirability of sending block diagram information to a remote computer for display, and so the Examiner's inclusion of this feature is not only unsupported in the cited art, but is clearly based on hindsight analysis, which is also improper.

(See Appeal Brief – page 15 to page 16, emphasis in original.)

Examiner's response:

o) Examiner disagrees. With respect to the Appellant's assertion that neither reference even hints at the desirability of sending block diagram information to a remote computer for display, the Examiner respectfully submits that the Examiner has addressed the Appellant's argument in the Examiner's responses (c), (e), and (n) hereinabove.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 59, 73, and 81 are proper and therefore, maintained.

Claims 65 and 74:

In the Appeal Brief, Appellant argues:

a) Cited col.9:31-50 describes sending REMOTE.HTM with references to a GIF file that represents the screen/interface for the server computer, referring to it as information regarding a graphical program, which is incorrect, at least because REMOTE.HTM and GIF file relate to

Huntsman's interface, not a block diagram. This text says nothing about a *graphical program* at all, much less sending information regarding a graphical program (nor, more specifically, a block diagram) to client computers over a network, nor, more particularly, where the information is usable by each client computer to display the graphical program or block diagram. A GUI is not a graphical program, as explained at length above, and thus Huntsman's sending of information for a GUI is not germane to sending information regarding a block diagram. Nor does combining Kodosky's block diagram with Huntsman's sending of GUI information remedy this deficiency—the combination still doesn't teach or suggest sending information regarding a block diagram over a network to a plurality of client computers for display on the client computers.

(See Appeal Brief – page 17, emphasis in original.)

Examiner's response:

a) Examiner disagrees. With respect to the Appellant's assertion that the combination of Huntsman and Kodosky doesn't teach or suggest sending information regarding a block diagram over a network to a plurality of client computers for display on the client computers, the Examiner respectfully submits that the Examiner has addressed the Appellant's argument in the Examiner's responses (a), (c), and (e) hereinabove.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 65 and 74 are proper and therefore, maintained.

Claims 68 and 75:

In the Appeal Brief, Appellant argues:

a) While Kodosky does disclose block diagrams, nowhere does Kodosky teach or suggest, or even hint at, (sending) information regarding a block diagram of a graphical program that is useable by a client computer system to display the block diagram on the web browser. In fact, Kodosky nowhere even mentions displaying a block diagram in a web browser at all. Applicant notes that Huntsman nowhere discloses or even hints at displaying a block diagram (nor any kind of program source code) on a web browser of a client computer system.

(See Appeal Brief – page 17 to page 18.)

Examiner's response:

a) Examiner disagrees. With respect to the Appellant's assertion that Kodosky nowhere even mentions displaying a block diagram in a web browser at all, the Examiner respectfully submits that the combination of Huntsman and Kodosky clearly discloses displaying a block diagram in a web browser. In the Office actions, the Examiner acknowledges that Huntsman does not disclose a graphical program including a block diagram as defined in the claims. Instead, Huntsman only discloses information describing the user interface and information regarding the graphical program that is useable by the client computer system to display the user interface and the graphical program on a web browser. Examiner relied upon Kodosky for its specific teaching of information regarding a block diagram of a graphical program. Thus, in view of the teaching of Kodosky and the teaching of Huntsman and the state of the art, one of ordinary skill in the art would be motivated to modify Huntsman's MS-Windows program (represented by the GIF file) as a block diagram of a graphical program of a virtual instrument in order to

remotely control block diagram information of a virtual instrument that is executing on one computer system from another computer system by displaying the block diagram information for a user to control.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 68 and 75 are proper and therefore, maintained.

Claims 69, 78, 90, and 100:

In the Appeal Brief, Appellant argues:

a) Cited col.9:50-57, 61-67, and col.10:1-6 describes a web browser determining mouse coordinates (and HTML mode variables) with respect to the presented GUI/screen, and sending these coordinates to the URL-addressed computer via hypertext- to-GUI-response means, but says nothing about a *block diagram* at all, much less receiving user input over the network editing a block diagram, and editing a block diagram accordingly. Nor does combining Kodosky's block diagram with Huntsman's sending of mouse coordinate and mode information remedy this deficiency, at least because Kodosky nowhere even hints at editing a block diagram remotely as claimed.

In the Response to Arguments, the Examiner argues that "Huntsman clearly discloses 'receiv[ing] user input specifying an edit to the graphical program from the client computer system over the network', citing Huntsman's remote GUI operations (col.10:1-20). However, Appellant notes that Huntsman never even mentions the term "edit", and nowhere describes editing any kind of program at all, much less a graphical program. Thus, Huntsman does not, and

cannot, provide the feature of remote editing of a program, and so, even in combination with Kodosky, which only teaches local editing of a block diagram, fails to teach or suggest these features of claim 69.

(See Appeal Brief – page 18, emphasis in original.)

Examiner's response:

a) Examiner disagrees. With respect to the Appellant's assertion that the combination of Huntsman and Kodosky says nothing about receiving user input over the network editing a block diagram, and editing a block diagram accordingly, the Examiner respectfully submits that the combination of Huntsman and Kodosky clearly discloses the claimed feature of receiving user input over the network editing a block diagram, and editing a block diagram accordingly. Huntsman clearly discloses "receiv[ing] user input specifying an edit to the graphical program from the client software over the network" *(see Column 9: 50-57, "The WWW browser, in accordance with HTML/HTTP protocol [9,6,7], will determine the coordinates pointed to be the mouse. The coordinates will be sent to the URL associated with the region in the map file, which will contain the address of the first computer. In addition to the coordinates, the HTML mode variables defined by the coordinated naming convention 5 will also be transmitted as the result of a click.")* and "edit[ing] the graphical program according to the user input specifying the edit" *(see Column 9: 61-67 to Column 10: 1-6, "The server control program 21 on the first computer 19 converts the HTML URL selection to GUI control commands using the hypertext-to-GUI-response means 7, and interpret the associated filename as a selection for the corresponding control according to the coordinated naming convention 5, and programmatically select the*

control or perform other action as request by the MODE and KEYTEXT variables using the programmatic-GUI-control-execution means 13 of the hypertext-to-GUI-response means 7."). However, Huntsman does not disclose specifying an edit to the block diagram of the graphical program. Examiner relied upon Kodosky for its specific teaching of specifying an edit to the block diagram of the graphical program *(see Column 18: 47-51, "FIG. 25 shows the EDIT menu selections ... CLEAR is useful for removing items from the active window, e.g., selected wires and structures from the block diagram window, or controls from the front panel window.").* Thus, in view of the teaching of Kodosky and the teaching of Huntsman and the state of the art, one of ordinary skill in the art would be motivated to incorporate the block diagram information of the GUI-based graphical program of a virtual instrument of Kodosky into Huntsman in order to allow the block diagram information of the virtual instrument to be remotely edited by a user.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 69, 78, 90, and 100 are proper and therefore, maintained.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the Examiner in the Related Appeals and Interferences section of this Examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Qing Chen

/Q. C./

Examiner, Art Unit 2191

Conferees:

/Wei Y Zhen/

Supervisory Patent Examiner, Art Unit 2191

/Lewis A. Bullock, Jr./

Supervisory Patent Examiner, Art Unit 2193